

Projection-Aware Task Planning and Execution for Human-in-the-Loop Operation of Robots in a Mixed-Reality Workspace

Tathagata Chakraborti, Sarath Sreedharan, Anagha Kulkarni, Subbarao Kambhampati

Abstract—Recent advances in mixed-reality technologies have renewed interest in alternative modes of communication for human-robot interaction. However, most of the work in this direction has been confined to tasks such as teleoperation, simulation or explication of individual actions of a robot. In this paper, we will discuss how the capability to project intentions affect the *task planning* capabilities of a robot. Specifically, we will start with a discussion on how projection actions can be used to reveal information regarding the future intentions of the robot at the time of task execution. We will then pose a new planning paradigm – *projection-aware planning* – whereby a robot can trade off its plan cost with its ability to reveal its intentions using its projection actions. Finally, we will show how in the context of task planning, projection actions may also be useful for plan explicability and explanations. We will demonstrate each of these scenarios with the help of a joint human-robot activity using the HoloLens.

I. INTRODUCTION

Effective planning for human robot teams not only requires the capacity to be “human-aware” during the plan generation process, but also the ability to interact with the human during the plan execution phase. Prior work has underlined this need [1] as well as explored ways to exchange [2] information in natural language during interaction with the human in the loop. However, the state of the art in natural language considerably limits the scope of such interactions, especially where precise instructions are required. In this project, we show how natural language can be replaced with the help of wearable technologies (such as the HoloLens) for the effective *communication of intentions* during proximal human-in-the-loop operation of robots.

The last decade has seen a massive increase in robots deployed on the factory floor [3]. This has led to fears of massive job loss for humans in the manufacturing industry, as well concerns of safety for the jobs that do remain. The latter is not an emerging concern, though. Automation of the manufacturing industry has gone hand in hand with incidents of misaligned intentions between the robots and their human co-workers, leading to at least four instances of fatality [4]. This dates back to as early as 1979 when a robot arm crushed a worker to death while gathering supplies in the Michigan Ford Motor Factory, to as recent as 2015 in a much publicized accident at the Volkswagen factory in Baunatal, Germany. With 1.3 million new robots predicted to enter the workspace by next year [5], such concerns are only expected

Parts of this project appeared in the U.S. Finals of the Microsoft Imagine Cup 2017. Details: <http://www.ae-robots.com/>

The authors are with the Department of Computer Science at Arizona State University, Tempe, AZ 85281 USA. Contact: { tchakra2, ssreedh3, akulkal16, rao } @ asu.edu

to escalate. A closer look at the dynamics of employment in the manufacturing industry, however, reveals that the introduction of automation has in fact increased productivity [6] as well as, surprisingly, contributed to a steady increase in the number of jobs for human workers [7] in Germany (which so far dominates in terms of deployed robots in the industry). We posit then either a semi-autonomous workspace in future with increased hazards due to misaligned interests of robots in the shared environment, or a future where the interests of the human workers will be compromised in favor of automation. In light of this, it is essential that the next-generation factory floor [8] is able to cope with the needs of these new technologies. Indeed, recent reports [9] have hinted at significant gains to be had from bridging this gap.

At the core of this problem is the impedance mismatch between humans and robots in how they represent and communicate information. Despite the progress made in natural language processing, natural language understanding is still a largely unsolved problem, and as such robots find it difficult to express their own goals and intentions effectively. Thus there exists a significant communication barrier to be overcome from either side, and robots are essentially still “autistic” [10] in many aspects. While this may not always be a serious concern for deploying completely autonomous agents in isolated environments such as for space or underwater exploration, the priorities change considerably when humans and robots are involved in collaborative tasks, especially for concerns of safety, if not to just improve the effectiveness of collaboration. This is emphasized in the *Roadmap for U.S. Robotics* [11] which outlines that “*humans must be able to read and recognize robot activities in order to interpret the robot’s understanding*”.

A. Related Work

The concept of intention projection for autonomous systems has, of course, been explored before. An early attempt was made in [12] in a prototype Interactive Hand Pointer (IHP) to control a robot in the human’s workspace. Similar systems have since been developed to visualize trajectories of mobile wheelchairs and robots [13], [14], which suggest that humans prefer to interact with a robot when it presents its intentions directly as visual cues. The last few years have seen active research [15], [16], [17], [18], [19], [20], [21], [22] in this area, but most of these systems were passive, non-interactive and quite limited in their scope, and did not consider the state of the objects or the context of the plan pertaining to the action while projecting information. As such, the scope of intention projection has remained largely

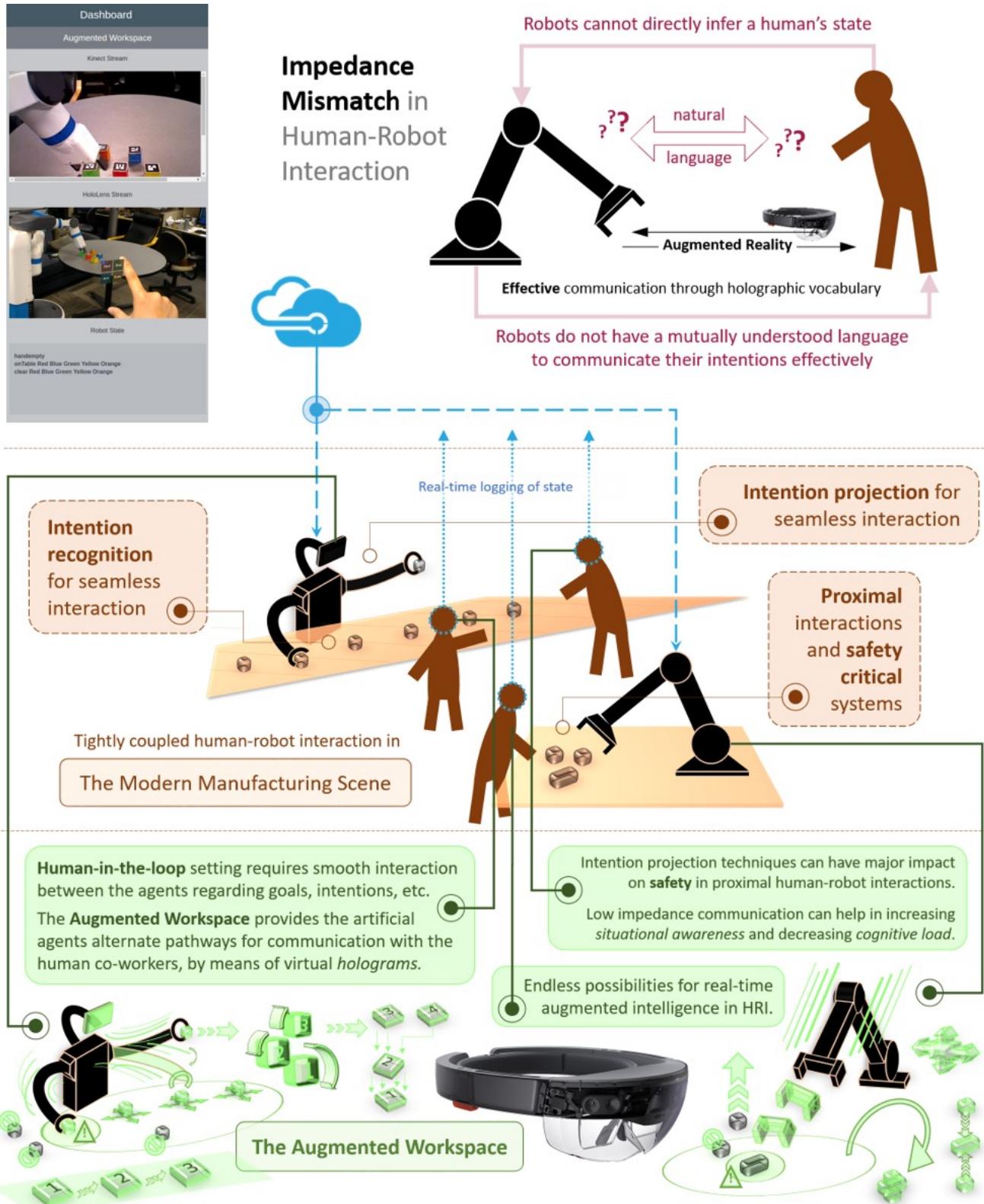


Fig. 1: A cloud-based distributed augmented workspace to combat impedance mismatch in human robot interactions.

limited. Indeed, recent works [23], [24], [25], [26] have made the first steps towards extending these capabilities to the context of task planning and execution, but fall short of formalizing the notion of intention projections beyond the current action under execution. Instead, in this paper, we demonstrate a system that is able to provide much richer information to the human-in-the-loop during collaborative plan execution, in terms of the current state information, action being performed as well as future parts of the plan under execution, particularly with the notion of explicating or foreshadowing future intentions. Recent advances [27] in the field of mixed reality make this form of online interactive plan explication particularly compelling.

B. Contributions

The contributions of our paper are three fold –

- In Sections IV and V we demonstrate how an *Augmented Workspace* can assist in *task-level* planning and execution in collaborative human-robot interactions.
- In Section IV we show how the intention projection techniques can be used to reduce **ambiguity over possible plans** during execution.
 - In Section IV-C we show how this consideration can be used to realize a *first of its kind* task planner that instead of considering only cost optimal plans, *generates* plans which are easier to explicate using intention projection actions.
- In Section V we demonstrate how the ability to project world information applies to the process of explanations for **inexplicability of a plan** during execution.

II. SYSTEM OVERVIEW

A. Collaborative Manufacturing / BlocksWorld Domain

Our primary focus here is on structured settings like the manufacturing environment where wearables can be a viable solution for improving the workspace. Indeed, a reboot of the safety helmet and goggles only requires retro-fitting existing wearables with sensors that can enable these new technologies. Imagine, then, a human and robot engaged in an assembly task, where they are constructing a structure collaboratively. Further suppose that the human now needs a tool from the shared workspace. At this time, neither agent is sure what tools and objects the other is going to access in the immediate future - this calls for seamless transfer of relevant information without loss of workflow. Existing (general purpose) solutions will suggest intention recognition [28] or natural language [2] communication as a means to respond to this situation. While natural language and intent or gesture recognition techniques remain the ideal choice in most cases, and perhaps the only choice in some (such as robots that would interact with people in their daily lives), we note that these are inherently noisy and ambiguous, and need not necessarily be the communication medium of choice in controlled environments such as on the factory floor or by the assembly line where the workspace can be engineered to enforce protocols in the interests of safety and

productivity, in the form of safety helmets integrated with wearable technology [29].

Instead, in our proposed system, the robot projects its intentions as *holograms* thus making them directly accessible to the human in the loop, e.g. by projecting a pickup symbol on a tool it might use in future. Further, unlike in traditional mixed reality projection systems, the human can directly interact with these holograms to make his own intentions known to the robot, e.g. by gazing at and selecting the desired tool thus forcing the robot to replan. To this end, we develop, with the power of the HoloLens, an alternative communication paradigm that is based on the projection of explicit visual cues pertaining to the plan under execution via holograms such that they can be intuitively understood and directly read by the human partner. The “real” shared human-robot workspace is now thus augmented with the virtual space where the physical environment is used as a medium to convey information about the intended actions of the robot, the safety of the work space, or task-related instructions. We call this the *Augmented Workspace*. Recent development of augmented reality techniques [30] has opened up endless possibilities in such modes of communication.

In this paper, we will use the classic [31] BlocksWorld domain as a proxy to a collaborative assembly domain. Here the robot is tasked with making words (or configurations) out of lettered (or colored) blocks using stacking and unstacking actions, to mimic assembly of specified structures. This will be used as the domain to illustrate various use cases and demonstrations in the rest of the paper.

B. System Architecture

Instead of the single human and robot collaborating over an assembly task, imagine now an entire workspace shared by many such agents, as is the case of most manufacturing environments. Traditional notions of communication become intractable in such settings. With this in mind, we make the entire system cloud based - all the agents log their respective states on to a central server, and can also access the state of their co-workers from it. As opposed to peer-to-peer information sharing, this approach provides a distinct advantage towards making the system scalable to multiple agents, both humans and robots, sharing and collaborating in the same workspace, as envisioned in Figure 1.

Apart from the Augmented Workspace which allows the robots to communicate with their human co-workers in the virtual space, the system conceptualizes a centralized *Dashboard* (Figure 1; top left inset) that provides a real-time snapshot of the entire workspace. The Dashboard allows the humans to communicate or visualize the collaborative planning process between themselves. It can be especially useful in factory settings to the floor manager who can use it to effectively monitor the shared workspace. It shows the real-time stream from the robots’ points of view, the AR streams from the humans’ points of view and information about the current status of plan execution.

Figure 2 provides a brief overview of the system architecture put in place to support the cloud based distributed

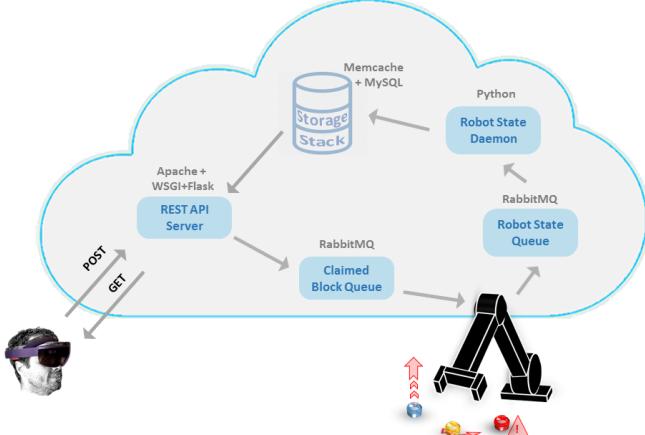


Fig. 2: Flow of control in the Augmented Workspace on the cloud-based infrastructure using restful APIs.

augmented workspace envisioned in Section I. In the augmented workspace, the HoloLens communicates with the user endpoints through the REST API server. The API server is implemented in python using the Flask web server framework. All external traffic to the server is handled by an Apache2 server that communicates with the python application through a WSGI middle layer. The Apache2 server ensures that the server can easily support a large number of concurrent requests.

The REST service exposes both GET and POST endpoints. The GET links provides the HoloLens application with a way of accessing information from the robot, while the POST link provides the HoloLens application control over the robots operation. Currently, we are using the API to expose information like the robotic planning state, robot joint values and coordinate transforms to markers in the environment. Most API GET calls will first try to fetch the requested information from the memcached layer, and would only try a direct query to the MySQL database if the cache entry is older than a specified limit. Each query to the database also causes the corresponding cache entry to be updated. The MySQL server is updated by a daemon that runs on Azure and keeps consuming messages sent from the robot through various queues implemented using the rabbitMQ service.

III. PRELIMINARIES OF AI PLANNING

In this section, we will introduce some of the preliminaries of automated planning and notations used in rest of the paper.

A Classical Planning Problem [32] is a tuple $\mathcal{M} = \langle \mathcal{D}, \mathcal{I}, \mathcal{G} \rangle$ with domain $\mathcal{D} = \langle F, A \rangle$ - where F is a set of fluents that define a state $s \subseteq F$, and A is a set of actions - and initial and goal states $\mathcal{I}, \mathcal{G} \subseteq F$. Action $a \in A$ is a tuple $\langle c_a, pre(a), eff^\pm(a) \rangle$ where c_a is the cost, and $pre(a), eff^\pm(a) \subseteq F$ are the preconditions and add/delete effects, i.e. $\delta_{\mathcal{M}}(s, a) \models \perp$ if $s \not\models pre(a)$; else $\delta_{\mathcal{M}}(s, a) \models s \cup eff^+(a) \setminus eff^-(a)$ where $\delta_{\mathcal{M}}(\cdot)$ is the transition function. The cumulative transition function is $\delta_{\mathcal{M}}(s, \langle a_1, a_2, \dots, a_n \rangle) = \delta_{\mathcal{M}}(\delta_{\mathcal{M}}(s, a_1), \langle a_2, \dots, a_n \rangle)$.

Note that the “model” \mathcal{M} of a planning problem includes the action model *as well as the initial and goal states of an agent*. The solution to \mathcal{M} is a sequence of actions or a (satisficing) plan $\pi = \langle a_1, a_2, \dots, a_n \rangle$ such that $\delta_{\mathcal{M}}(\mathcal{I}, \pi) \models \mathcal{G}$. The cost of a plan π is $C(\pi, \mathcal{M}) = \sum_{a \in \pi} c_a$ if $\delta_{\mathcal{M}}(\mathcal{I}, \pi) \models \mathcal{G}$; ∞ otherwise. The cheapest plan $\pi^* = \arg \min_{\pi} C(\pi, \mathcal{M})$ is the (cost) optimal plan. We refer to this cost as \mathcal{M} as C^*_M .

Projection actions can include information regarding either the state of the world or the robot’s plans – both of these can reveal information regarding the robot’s future intentions, i.e. goals or plans. In this work, we assume a very simple projection model based on the truth value of specified conditions in parts of the plan yet to be executed –

An Action Projection AP is defined as a mapping $u : [0..|\pi|] \times \mathcal{A} \mapsto \{\text{T}, \text{F}\}$ indicating $\exists j \geq i$ where $a_i, a_j \in \pi$ if $u(j, a_i) = \text{T}$ and $a_i, a_j \notin \pi$ otherwise – i.e. existence or membership of an action in the rest of the plan.

A State Value Projection SVP is defined as a mapping $v : \mathcal{F} \times \mathcal{A} \mapsto \{\text{T}, \text{F}\}$ so that there exists a state in the state sequence induced by the sub-plan starting from a_i where the state variable $f \in \mathcal{F}$ holds the value $v(f, a_i)$, i.e. $\exists s' : \delta_{\mathcal{M}}(s, \pi') \models s'$ where s is the current state and π' is the sub-plan $(\pi)_{k \leq |\pi|}^{k \leq |\pi|}$ and $f \in s$ iff $v(f, a_i) = \text{T}$, $f \notin s$ otherwise.

In the following sections, we will discuss how a robot can determine *when* to deploy *which* of these projections in order to better explicate its plans to a human in the loop.

IV. PROJECTIONS FOR AMBIGUOUS INTENTIONS

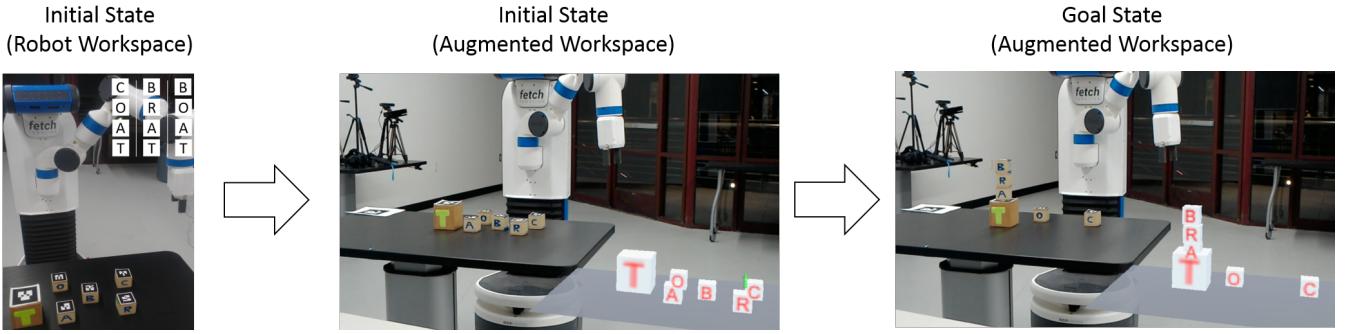
In this section, we will concentrate upon how projection actions can resolve ambiguity with regards to the intentions of a robot in the course of execution of a task plan.

A. Projection-Aware Plan Execution

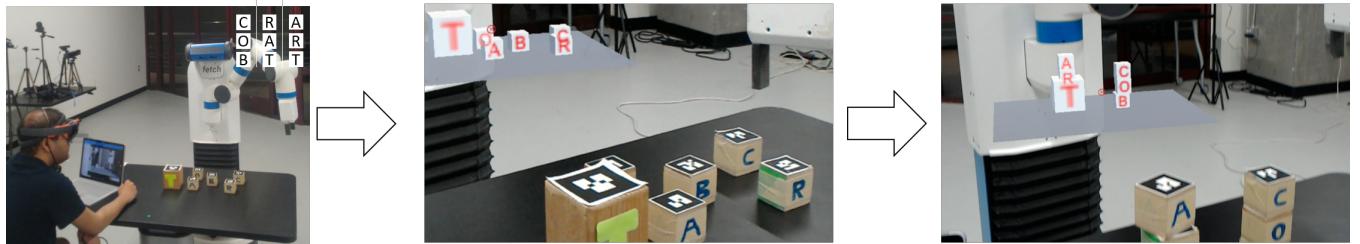
The first topic of consideration is the projection of intentions of a robot with a human *observer* in the loop.

Illustrative Example. Consider a robot involved in a mock assembly task (i.e. block stacking) as shown in Figure 3a. We will be using this setting throughout the rest of the paper. Here, the robot’s internal goal is to form the word BRAT. However, given the letters available to it, it can form other words as well – consider two more possible goals BOAT and COAT. As such, it is difficult to say, from the point of view of the observer, by looking at the starting configuration, which of these is the real outcome of the impending plan. The robot can, however, at the start of its execution, choose to indicate that it has planned to pick up the block R later (by projecting a bobbing arrow on top of it), thereby resolving this ambiguity. A video demonstrating this can be viewed at <https://goo.gl/SLgCPE>.

A Projection-Aware Plan Execution Problem PAPEP is defined as the tuple $\Phi = \langle \pi, \Pi, \{AP\}_i, \{SVP\}_i \rangle$ where π is the robot’s plan up for execution, Π (which includes π) is the set of possible plans it can execute, and $\{AP\}_i$ and $\{SVP\}_i$ are the set of action and state value projections available to it (i.e. induced by π).



(a) The robot projects (AP) a green arrow to indicate pickup of a block that is part of an optimal plan to only one of its possible goals.



(b) The robot inverts the projection context and this time shows (SVP) which block is *not* going to be available using a red cross.

Fig. 3: Projection-Aware Plan Execution for human observer and human-in-the-loop scenarios.

The solution to Φ is a composite plan $\pi^c \circ \pi$ where $\pi^c \subseteq \{AP\}_i \cup \{SVP\}_i$ are the projection actions that disambiguate the plans at the time of execution. We compute this using the concept of resource profiles, as introduced in [33]. Informally, a **resource** [33] is defined as any state variable whose binary value we want to track. We will use this concept to tie each action or state value projection action to a single resource variable, whose effect can be monitored. For example, a *not-clear* predicate will indicate that a block is in use or not available while an action that produces or negates that predicate – e.g. *pick-up* can be similarly tracked through it. This mapping between projection actions and their corresponding resource variables is domain-dependent knowledge that is provided up-front.

A Resource Profile \mathcal{R}^π induced by a plan π on a resource r is a mapping $\mathcal{R}^\pi : [0..|\pi|] \times r \mapsto \{0, 1\}$, so that r is *locked* [33] by π at step i if $\mathcal{R}^\pi(r, i) = 1$ and it is free otherwise.

A Cumulative Resource Profile \mathbb{R}^Π induced by a set of plans Π on a resource r is a mapping $\mathbb{R}^\Pi : [0.. \max_{\pi \in \Pi} |\pi|] \times r \mapsto [0, 1]$, so that r is *locked* with a probability $\mathbb{R}^\Pi(r, i) = \sum_{\pi \in \Pi} \mathcal{R}^\pi(r, i) \times P(\pi)$, where $P(\pi)$ is the prior probability of plan π (assumed uniform in this paper).

The set of projection actions π^c in the solution to the PAPEP Φ are found by computing –

$$\arg \min \sum_r \sum_{\pi \in \Pi} \mathcal{R}^\pi \times \mathbb{R}^\Pi \quad (1)$$

Thus, we are minimizing the conflicts between the current plan and other possible plans, so that the projection actions that are tied to the resources with the minimal conflicts give us the most distinguishing projection. This is iterated over, in case none of the projections have zero conflicts.

B. Projection-Aware Human-in-the-Loop Plan Execution

In the previous example, we confined ourselves to situations with the human only as the observer. Now, we consider a situation where both the human and the robot are involved in task planning in a collaborative sense, i.e. both the human and the robot perform actions in a joint plan to achieve their goals which may or may not be shared.

Illustrative Example. Going back to the running example of the block stacking task, now consider that the robot and the human both have goals to make a three letter word out of ART, RAT and COB (as seen in Figure 3b). The robot has decided to make the word ART, but realizes that this leaves the human undecided on how to proceed. Thus the disambiguating projection action here includes annotating the R block with a “not available” symbol so that the only possible goal left for the human is COB. A video demonstrating this can be viewed at <https://goo.gl/SLgCPE> (same as in Section IV-A). Note that in this case the robot, in coming up with a useful projection action, has reversed the perspective from what is relevant to its own plan, to information that negates possible plans of the human in the loop.

A Projection-Aware Human-in-the-Loop Plan Execution Problem PAHILPEP is defined as the tuple $\Psi = \langle \Pi^R, \Pi^H, \mathbb{G}, \{\text{AP}\}_i, \{\text{SVP}\}_i \rangle$ where Π^R and Π^H are the set of possible plans the robot and the human can execute respectively, \mathbb{G} is their shared team goal, and $\{\text{AP}\}_i$ and $\{\text{SVP}\}_i$ are the set of action and state value projections available to the robot (i.e. induced by Π^R).

The solution to Φ is, as before, a composite plan $\pi^c \circ \pi^R$ where the projection actions are composed with the robot’s component of the joint team plan, such that $\delta(\mathcal{I}, \pi^c \circ \pi^R \circ$

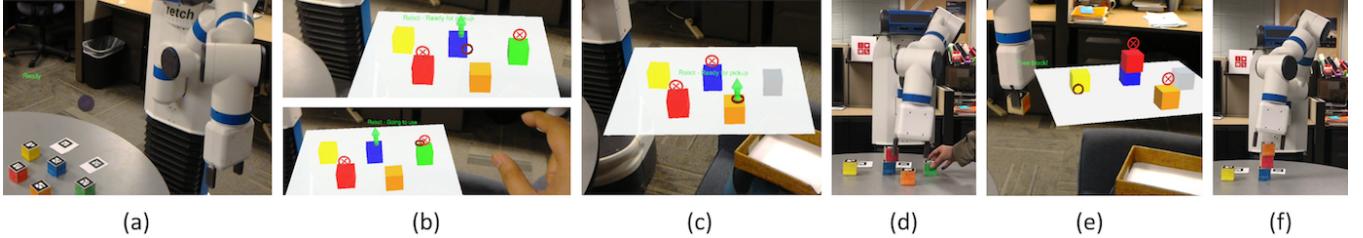


Fig. 4: Interactive execution of a plan in the Augmented Workspace - (a) the robot wants to build a tower of height three with blocks blue, red and green. (b) Blocks are annotated with intuitive holograms, e.g. an upward arrow on the block the robot is going to pick up immediately and a red cross mark on the ones it is planning to use later. The human can also gaze on an object for more information (in the rendered text). (c) & (d) The human pinches on the green block and claims it for himself. The robot now projects a faded out green block and re-plans online to use the orange block instead (as evident by pickup arrow that has shifted on the latter at this time). (e) Real-time update and rendering of the current state showing status of the plan and objects in the environment. (f) The robot completes its new plan using the orange block.

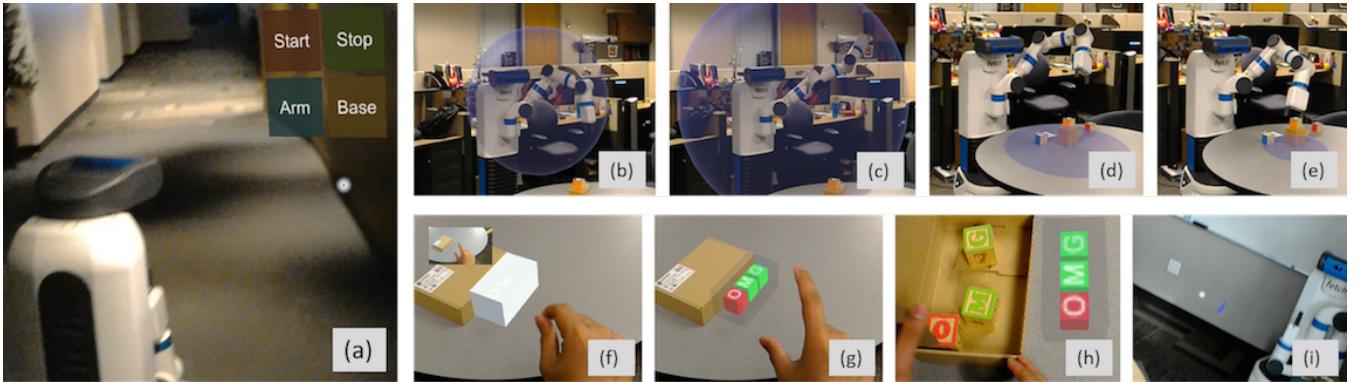


Fig. 5: Interactive plan execution using the (a) Holographic Control Panel. Safety cues showing dynamic real-time rendering of volume of influence (b) - (c) or area of influence (d) - (e), as well as (i) indicators for peripheral awareness. Interactive rendering of hidden objects (f) - (h) to improve observability and situational awareness in complex workspaces.

$\pi^H) \models \mathbb{G}$. The set of projection actions π^c in the solution to the PAHILPEP Ψ is again found by computing –

$$\arg \max \sum_r \sum \mathbb{R}^{\Pi^H} \times \mathbb{R}^{\Pi^R} \quad (2)$$

Notice the inversion to `argmax`, since in the case of an active human in the loop, so as to provide the most pertinent information regarding conflicting intentions to the human.

Remark. The use of joint plans [34] to reason over different modes of human-robot interactions has been investigated before, particularly in the context of using resource profiles [33] for finding conflicts in the human’s and the robot’s plans. It is interesting to note the reversed dynamics of interaction in the example provided above – i.e. in [33] the resource profiles were used so that the robot could replan based on probable conflicts so as to preserve the expected plans of the human. Here, we are using them to identify information to project to the human, so that the latter can replan instead.

1) *Closing the Loop – Replanning:* Of course, it may not be possible to always disentangle plans completely towards achievement of a shared goal in a collaborative setting. In the next demonstration, we show how the communication loop is closed by allowing the humans to interact directly

with the holograms in the augmented workspace and spawn replanning commands to be handled by the robot, in the event of conflicting intentions. For example, as in the previous examples, a robot involved in a collaborative assembly task can project the objects it is intending to manipulate into the human’s point of view, and annotate them with holograms that correspond to intentions to use or pickup. The human can, in turn, access or claim a particular object in the virtual space and force the robot to re-plan, without there ever being any conflict of intentions in the real space. The humans in the loop can thus not only infer the robot’s intent immediately from these holographic projections, but can also interact with them to communicate their own intentions directly and thereby modify the robot’s behavior online. The robot can also then ask for help from the human, using these holograms. Figure 4 demonstrates one such scenario. The human can also go into finer control of the robot by accessing the Holographic Control Panel, as seen in Figure 5(a). The panel provides the human controls to start and stop execution of the robot’s plan, as well as achieve fine grained motion control of both the base and the arm by making it mimic the user’s arm motion gestures on the MoveArm and MoveBase holograms attached to the robot.

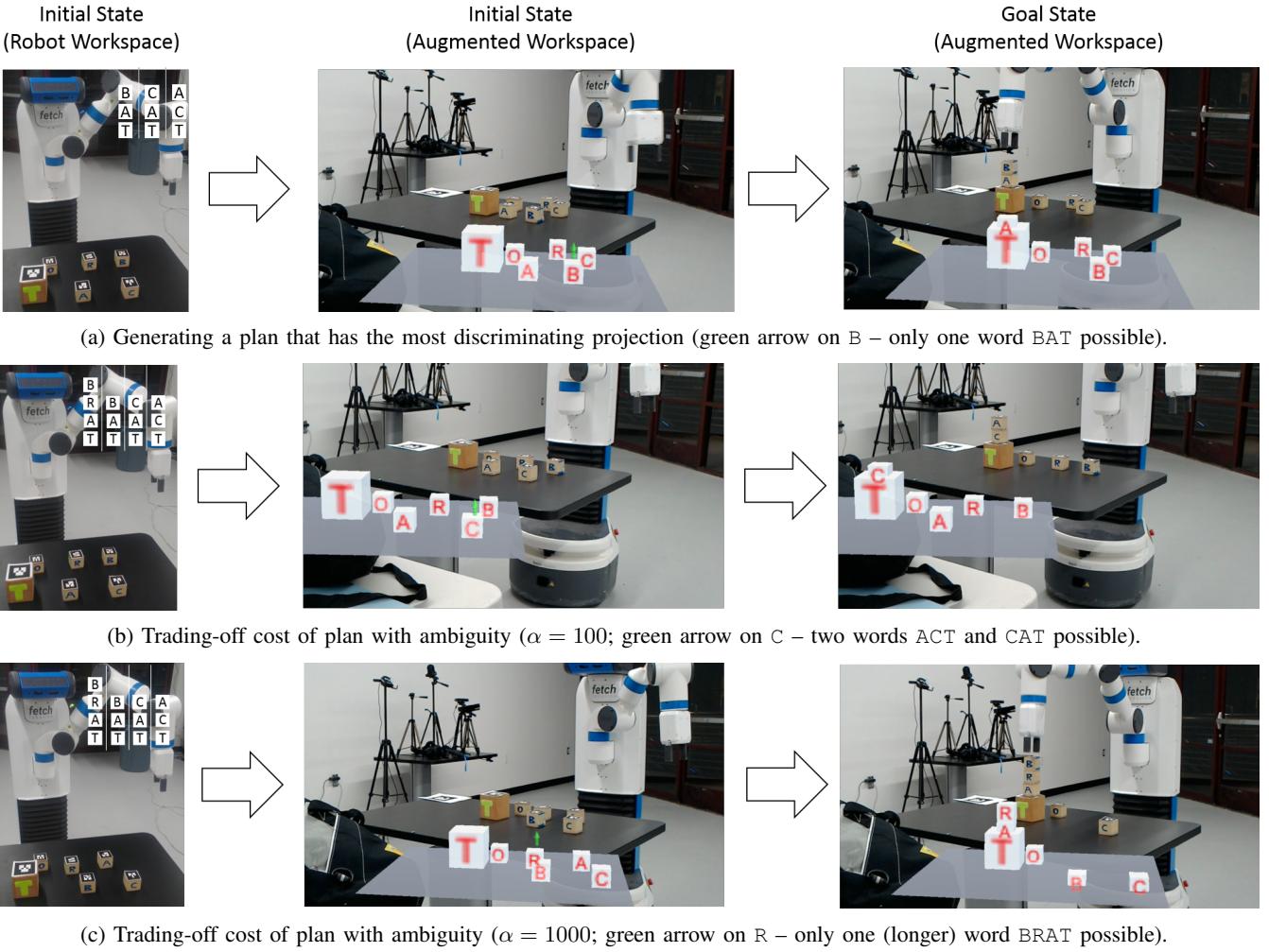


Fig. 6: Projection-aware plan generation illustrating trade-off in plan cost and goal ambiguity during execution.

2) *Closing the Loop – Assistive Cues:* The use of AR is, of course, not just restricted to procedural execution of plans. It can also be used to annotate the collaborative workspace with artifacts derived from the current plan under execution in order to improve the fluency of collaboration. For example, Figure 5(b-e) shows the robot projecting its area of influence in its workspace either as a 3D sphere around it, or as a 2D circle on the area it is going to interact with. This is rendered dynamically in real-time based on the distance of the end effector to its center, and to the object to be manipulated. This can be very useful in determining safety zones around a robot in operation. As seen in Figure 5(f-i), the robot can also render hidden objects or partially observable state variables relevant to a plan, as well as indicators to improve peripheral vision of the human, to improve their situational awareness.

Demonstrations for Sections IV-B.1 and IV-B.2 can be viewed at <https://goo.gl/pWWzJb>.

C. Projection-Aware Plan Generation

Now that we have demonstrated that intention projection can be an effective tool to disambiguate possible tasks at the time of the execution of a task plan, we ask *is it possible to*

use the knowledge of this ability to generate plans that are easier to disambiguate in the first place?

Illustrative Example. Consider again the blocks stacking domain, where the robot is yet to decide on a plan, but it has three possible goals BAT, CAT and ACT (as seen in Figure 6a). From the point of view of cost optimal planning, all these are equally good options. However, notice that the letter B is in only one of the words, while the others are in at least two possible words. Thus the robot is able to *reduce the ambiguity in plans* and decide to make the word BAT over the other options as a means of achieving the goal of making a word from the given set.

Illustrative Example. Now imagine that we have extended the possible set of words { BAT, CAT, ACT } with a longer word BRAT. The robot responds by projecting R and completes this longer word now, given R is the most discriminating action, and the possibility of projecting it ahead completely reveals its intentions **even though it involves the robot doing a longer and hence costlier plan** as seen in Figure 6c. This trade-off in the cost of plans and the ambiguity of intentions forms the essence of what we refer to as *projection-aware*

planning. In fact, we can show that by correctly calibrating this trade-off, we can achieve different sweet spots in how much the robot decides to foreshadow disambiguating actions. As seen in Figure 6b, in cases where the action costs are relatively greater than gains due to resolved ambiguity, the robot achieves a middle-ground of generating a plan that has the same cost as the optimal plan to achieve the goal of making a word from this set, but also involves reasonable forecasting of (two) possible goals by indicating a future pick-up action on C. A video demonstrating these behaviors can be viewed at <https://goo.gl/bebtWS>.

A Projection-Aware Planning Problem PAPP is defined as the tuple $\Lambda = \langle \mathcal{M}, \kappa, \{AP\}_i, \{SVP\}_i \rangle$ where \mathcal{M} is a planning problem and κ is a set of disjunctive landmarks.

The solution to Λ is a plan such that –

- π achieves the goal; and
- commitments imposed by the projection actions, i.e. future state conditions indicated by SVPs or actions promised by APs, as described in Section III are respected in the final plan.

The search for which projection actions to include is achieved by modifying a standard A-star search¹ [36], [32] so that the objective function trades off actions costs and ambiguity of future plans (to possible landmarks) from the current state (as shown in Algorithm 1). This is given by –

$$\alpha C(\hat{\pi}, \mathcal{M}) + \beta \sum \mathbb{E}(\Pi) \quad (3)$$

Here Π is a set of possible plans that the robot can pursue from the current state. These are the top-K plans [37] to the landmarks [38] in the domain². $\mathbb{E}(\Pi)$ is the entropy of the probability distribution [39] over the plan set Π given the current plan prefix $\hat{\pi}$ to that state. Since a full evaluation of the plan recognition problem in every node is prohibitively expensive, we use a simple observation model where the currently proposed projection action tests *membership* of its parent action if it is an AP (or state value if it is an SVP) in the delete-relaxed plans [40] to each landmark –

$$\alpha C(\pi, \mathcal{M}) + \beta \sum_{\kappa} I(a_i \in \pi - del) \quad (4)$$

where I is the indicator function indicating if the current action a_i is part of the delete-relaxed plan $\pi - del$ from the current state to each of the landmarks κ . The details^{3,4} of the search are provided in Algorithm 1. Notice that the indicator function only comes into play when projection

¹Note that to speed up search we used “outer entanglement” analysis [35] to prune unnecessary actions for the blocks stacking domain.

²In our demonstration, the plan set was composed of the optimal plan (top-1) to the landmarks induced by the all possible words in the domain needed to reach the goal of forming any word.

³We currently handle only APs in the solution to a PAPP. Also, the number of APs in a solution were restricted to a maximum of two to three due to the time consuming nature of computing Π . This can be sped up by precomputing the relaxed planning graph.

⁴We will use the shorthand notation $AP(a) = \langle c, \phi, \phi \rangle$ to refer to the projection action corresponding to an action $a \in \mathcal{A}$.

Algorithm 1 Projection-Aware Planning Algorithm

```

1: procedure PAPP-SEARCH
2:   Input: PAPP  $\Lambda = \langle \mathcal{M}, \kappa, \{AP\}_i, \phi \rangle$ 
3:   Output: Plan  $\pi$ 
4:   Procedure:
5:    $\mathcal{A} \leftarrow \mathcal{A} \cup \{AP\}_i$                                 ▷ Add projections to action set
6:   fringe  $\leftarrow$  Priority_Queue()
7:   fringe.push( $\langle \mathcal{I}, \langle \rangle \rangle, 0$ )
8:   while True do
9:      $\langle \hat{S}, \hat{\pi} \rangle, c \leftarrow$  fringe.pop()
10:    if goal check true then return  $\hat{\pi}$                                 ▷ Refer to Section IV-C
11:    else
12:      for  $a \in \mathcal{A}$  do
13:        if  $\hat{s} \models pre(a)$  then
14:           $\hat{s}' \leftarrow \delta(\hat{s}, a)$ 
15:          fringe.push( $\langle \hat{s}', \hat{\pi} + a \rangle, F(\hat{s}', a, \hat{\pi})$ )
16: procedure  $F(\hat{s}', a, \hat{\pi})$ 
17:   if  $a \notin \{AP\}_i$  and  $AP(a) \notin \hat{\pi}$  then
18:     return  $c_a + C(\hat{\pi}, \mathcal{M})$ 
19:   else
20:     if  $a \in \{AP\}_i$  then
21:       Compute  $\Pi = \{top - K \text{ delete - relaxed plans to } \kappa\}$ 
22:        $N \leftarrow 0$ 
23:       for  $\pi \in \Pi$  do
24:         if  $AP^{-1}(a) \in \pi$  then
25:            $N \leftarrow N + 1$ 
26:       return  $\alpha(c_a + C(\hat{\pi}, \mathcal{M})) + \beta N$                                 ▷ (Equation 4)
27:     else
28:       return  $\gamma c_a + C(\hat{\pi}, \mathcal{M})$ 

```

actions are being pushed into the queue, thus biasing the planner towards producing plans that are easier to identify based on the projections. Also note how the cost of actions corresponding to projections in the plan prefix are discounted by a factor γ (to be set depending on how much the designer wants to incentivize projections).

Remark. We note that the ability to communicate information, and planning with the knowledge of that ability when it is useful to disambiguate intentions, is not necessarily unique to mixed-reality interactions only. One could use the planner introduced in Section IV-C to generate *content* for traditional speech-based interactions as well (c.f. recent works on verbalization of intentions in natural language [2], [41]). However, as demonstrated in this paper, the medium of mixed-reality provides a particularly concise and effective alternative vocabulary of communication, especially in more structured scenarios such as in collaborative manufacturing.

Remark. Recent work in the scope of human-aware task and motion planning has focused on generation of legible motion plans [42], [43] and explicable task plans [44], [45] with the notion of trading off cost of plans with how easy they are to interpret for a human observer. This runs parallel to our work on planning with intention projections. Note that, in effect, either during the generation or the execution of a plan, we are, in fact, trying to optimize the same criterion. However, in our case, the problem becomes much more intriguing since the robot gets to *enforce legibility or explicability of a plan by foreshadowing of actions that have not been executed yet*. Indeed, this connection has also been hinted at in recent work [46]. However, to the best of our knowledge, this is the first task-level planner to achieve this trade-off.

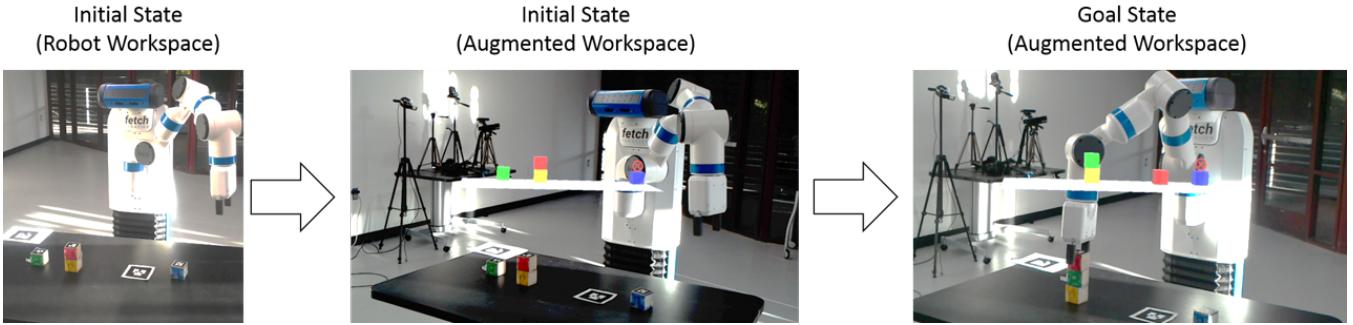


Fig. 7: The human has instructed the robot to make a tower of height 3 with the red block on top. Since the blue block is not reachable it has to unstack red in order to achieve its goal. This is a suboptimal plan to the observer who may not know the robots internal trajectory constraints. The robot thus decides to project a red error symbol on the blue block indicating it is not reachable. The optimal plans in both models now align.

In relation to plan explicability, we note that the plan explanations and explicability process forms a delicate balancing act, as investigated in recent work [47]. This also has interesting implications to the intention projection ability. This is the topic of the next section.

V. PROJECTIONS FOR INEXPLICABLE ACTIONS

In the previous section, we had focused on dealing with *ambiguity* of intentions during execution of a plan. Now we will deal with *inexplicability* of actions, i.e. how to use projection capabilities to annotate parts of the world so that a plan under execution “makes sense” to the observer.

Illustrative Example. Going back to our block stacking setting, consider a scenario where the human-in-the-loop asks the robot to make a tower of height three with the red block on top (please refer to the attached supplementary video file). Here the optimal plan from the point of view of the observer is likely to be as follow –

```
>> Explicable Plan
>> action :: pick-up green
>> action :: stack green blue
>> action :: pick-up red
>> action :: stack red green
```

However, not all the blocks are reachable, as determined by the internal trajectory constraints of the robot. So the optimal plan for the robot would instead be longer –

```
>> Robot Optimal Plan
>> action :: pick-up red
>> action :: put-down red
>> action :: pick-up yellow
>> action :: stack yellow blue
>> action :: pick-up red
>> action :: stack red blue
```

This plan is, of course, inexplicable if the observer knows that the robot is a rational agent, given the former’s understanding of the robot model. The robot can chose to mitigate this situation by annotating the unreachable blocks as “*not reachable*” as shown in Figure 7. A video demonstration can be seen at <https://goo.gl/TRZcW6>.

The identification of projection actions in anticipation of inexplicable plans closely follows the notion of multi-model explanation generation algorithms studied in [32].

A **Multi-Model Planning Problem** [32] is the tuple $\Gamma = \langle \mathcal{M}^R, \mathcal{M}_h^R \rangle$ where $\mathcal{M}^R = \langle D^R, \mathcal{I}^R, \mathcal{G}^R \rangle$ and $\mathcal{M}_h^R = \langle D_h^R, \mathcal{I}_h^R, \mathcal{G}_h^R \rangle$ are respectively the planner’s model of a planning problem and the human’s understanding of it.

In our block stacking domain, multiple models are spawned due to internal constraints of the robot that the human may not be aware of (e.g. reachability) while the world model (i.e. how the world works - the robot has to pick up and object to put it down, etc.) is shared across both the models. As these models diverge, plans that are optimal in the robot’s model may no longer be so in the human’s and thus become *inexplicable*. The robot can mitigate these situation by generating *multi-model explanations* studied extensively [32], [48], [49] in recent literature.

A **Multi-Model Explanation** [32] is a solution to an MMP in the form of a model update to the human so that the optimal plan in the robot’s model is now also optimal in the human’s updated model. Thus, a solution to Γ involves a plan π and an explanation \mathcal{E} such that –

- (1) $C(\pi, \mathcal{M}^R) = C_{\mathcal{M}^R}^*$;
- (2) $\widehat{\mathcal{M}}_h^R \leftarrow \mathcal{M}_h^R + \mathcal{E}$; and
- (3) $C(\pi, \widehat{\mathcal{M}}_h^R) = C_{\widehat{\mathcal{M}}_h^R}^*$.

We use the same to generate content for the explanations conveyed succinctly through the medium of mixed reality, as described in the illustrative example above.

VI. CONCLUSION

In conclusion, we showed how an augmented workspace can be used to improve collaboration among humans and robots from the perspective of task planning. This can be either in terms of an interactive plan execution process where the robot can foreshadow future actions to reveal its intentions, or in the context of a *projection-aware* plan generation process where the robot can trade-off the ambiguity in its intentions from the perspective of the human in the loop with the cost of plans. Finally, we showed how explanatory

“dialogs” with the human as a response to inexplicable plans can be conducted in this mixed-reality medium.

Such modes of interaction open up several exciting avenues of future research. Particularly, as it relates to task planning, we note that while we had encoded some of the notions of ambiguity in the planning algorithm itself, the vocabulary of projections can be much richer and as such existing representations fall short of capturing or exploiting these relationships (e.g. action X is going to happen 3 steps after action Y). A holographic vocabulary thus calls for the development of representations – PDDL3.x – that can capture such complex interaction constraints modeling not just the planning ability of the agent but also its interactions with the human. Further, such representations can be *learned* to generalize to methods that can, given a finite set of symbols or vocabulary, compute domain independent projection policies that decide what and when to project to reduce cognitive overload on the human.

Video Demonstrations. Demonstrations of all the use cases in the paper can be viewed at <https://goo.gl/Gr47h8>. The code base for the projection-aware plan generation and execution algorithms are available at <https://github.com/TathagataChakraborti/ppap>.

Acknowledgements. This research is supported in part by the AFOSR grant FA9550-18-1-0067, the ONR grants N00014161-2892, N00014-13-1-0176, N00014- 13-1-0519, N00014-15-1-2027, and the NASA grant NNX17AD06G. The first author is also supported by the IBM Ph.D. Fellowship from 2016-18. We would also like to thank Professors Heni Ben Amor and Yu “Tony” Zhang (Arizona State University) for their valuable inputs during the project.

REFERENCES

- [1] E. Karpas, S. J. Levine, P. Yu, and B. C. Williams, “Robust execution of plans for human-robot teams,” in *ICAPS*, 2015, pp. 342–346.
- [2] S. Tellex, R. Knepper, A. Li, D. Rus, and N. Roy, “Asking for help using inverse semantics,” in *RSS*, 2014.
- [3] Robotonomics, “Industrial robot,” 2017.
- [4] S. Weiss, “Could your robot hurt you? perhaps, but not intentionally.” Brown Human-Robot Interaction (HRI) Lab., 2015.
- [5] PRNewswire, “Global survey: 1.3 million industrial robots to enter service by 2018,” The International Federation of Robotics., 2016.
- [6] M. Muro and S. Andes, “Robots seem to be improving productivity, not costing jobs,” Harvard Business Review., 2015.
- [7] C. Look, “Robots are coming, but not for your job,” Bloomberg., 2016.
- [8] Automation World, “President Barack Obama and German Chancellor Angela Merkel visit Weidmullers Industry 4.0 Cockpit at Hannover Messe.” <https://bit.ly/2ICUjD5>, 2016.
- [9] Linette Lopez, “The robots are killing Tesla,” <https://read.bi/2IcOxrw>, 2018, business Insider.
- [10] G. A. Kaminka, “Curing robot autism: a challenge,” in *AAMAS*, 2013.
- [11] H. I. Christensen, T. Batzinger, K. Bekris, K. Bohringer, J. Bordogna, G. Bradski, O. Brock, J. Burnstein, T. Fuhlbrigge, R. Eastman *et al.*, “A roadmap for us robotics: from internet to robotics,” *Computing Community Consortium and Computing Research Association*, 2009.
- [12] S. Sato and S. Sakane, “A human-robot interface using an interactive hand pointer that projects a mark in the real work space,” in *ICRA*, vol. 1. IEEE, 2000, pp. 589–595.
- [13] A. Watanabe, T. Ikeda, Y. Morales, K. Shinozawa, T. Miyashita, and N. Hagita, “Communicating robotic navigational intentions,” in *IROS*. IEEE, 2015, pp. 5763–5769.
- [14] R. T. Chadalavada, H. Andreasson, R. Krug, and A. J. Lilienthal, “That’s on my mind! robot to human intention communication through on-board projection on shared floor space,” in *ECMR*, 2015.
- [15] S. Omidshafiei, A.-A. Agha-Mohammadi, Y. F. Chen, N. K. Ure, J. P. How, J. Vian, and R. Surati, “Mar-cps: Measurable augmented reality for prototyping cyber-physical systems,” in *AIAA ARC*, 2015.
- [16] S. Omidshafiei, A.-A. Agha-Mohammadi, Y. F. Chen, N. K. Ure, S.-Y. Liu, B. T. Lopez, R. Surati, J. P. How, and J. Vian, “Measurable augmented reality for prototyping cyberphysical systems: A robotics platform to aid the hardware prototyping and performance testing of algorithms,” *IEEE Control Systems*, vol. 36, no. 6, pp. 65–87, 2016.
- [17] J. Shen, J. Jin, and N. Gans, “A multi-view camera-projector system for object detection and robot-human feedback,” in *ICRA*, 2013.
- [18] K. Ishii, S. Zhao, M. Inami, T. Igarashi, and M. Imai, “Designing laser gesture interface for robot control,” in *INTERACT*, 2009.
- [19] P. Mistry, K. Ishii, M. Inami, and T. Igarashi, “Blinkbot: look at, blink and move,” in *UIST*. ACM, 2010, pp. 397–398.
- [20] F. Leutert, C. Herrmann, and K. Schilling, “A spatial augmented reality system for intuitive display of robotic data,” in *HRI*, 2013.
- [21] M. Turk and V. Fragoso, “Computer vision for mobile augmented reality,” in *Mobile Cloud Visual Media Computing*, 2015.
- [22] I. Maurtua, N. Pedrocchi, A. Orlandini, J. de Gea Fernández, C. Vogel, A. Geenen, K. Althoefer, and A. Shafti, “Fourbythree: Imagine humans and robots working hand in hand,” in *Emerging Technologies and Factory Automation (ETFA), 2016 IEEE 21st International Conference on*. IEEE, 2016, pp. 1–8.
- [23] R. S. Andersen, O. Madsen, T. B. Moeslund, and H. B. Amor, “Projecting robot intentions into human environments,” in *RO-MAN*, 2016, pp. 294–301.
- [24] Ramsundar Kalpagam Ganesan, “Mediating Human-Robot Collaboration through Mixed Reality Cues,” *M.S. Thesis*, 2017, Arizona State University.
- [25] Yash K. Rathore, “Facilitating Human-Robot Collaboration Using a Mixed-Reality Projection System,” *M.S. Thesis*, 2017, Arizona State University.
- [26] T. Chakraborti, S. Sreedharan, A. Kulkarni, and S. Kambhampati, “Alternative modes of interaction in proximal human-in-the-loop operation of robots,” *CoRR*, vol. abs/1703.08930, 2017, appeared in the U.S. Finals of the 2017 Microsoft Imagine Cup. Also appeared in ICAPS 2017 Workshop on User Interfaces for Planning and Scheduling and the ICAPS 2017 System Demonstrations Track.
- [27] T. Chakraborti, T. Williams, D. Szafir, and H. B. Amor, Eds., *Proceedings of the First Workshop on Virtual, Augmented and Mixed Reality for Human-Robot Interaction (VAM-HRI)*. HRI, 2018.
- [28] B. Hayes and B. Scassellati, “Autonomously constructing hierarchical task networks for planning and human-robot collaboration,” in *ICRA*. IEEE, 2016, pp. 5469–5476.
- [29] E. Ruffaldi, F. Brizzi, F. Tecchia, and S. Bacinelli, “Third point of view augmented reality for robot intentions visualization,” in *AVR*. Springer, 2016, pp. 471–478.
- [30] Sean O’Kane, “Microsoft used this adorable robot to show off new hololens features.” The Verge., 2015.
- [31] International Planning Competition, “IPC Competition Domains,” <https://goo.gl/i35bxc>, 2011.
- [32] T. Chakraborti, S. Sreedharan, Y. Zhang, and S. Kambhampati, “Plan Explanations as Model Reconciliation: Moving Beyond Explanation as Soliloquy,” in *IJCAI*, 2017.
- [33] T. Chakraborti, Y. Zhang, D. Smith, and S. Kambhampati., “Planning with resource conflicts in human-robot cohabitation,” in *AAMAS*, 2016.
- [34] T. Chakraborti, G. Briggs, K. Talamadupula, Y. Zhang, M. Scheutz, D. Smith, and S. Kambhampati, “Planning for serendipity,” in *Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on*, 2015, pp. 5300–5306.
- [35] L. Chrpa and R. Barták, “Reformulating planning problems by eliminating unpromising actions.” 2009.
- [36] P. E. Hart, N. J. Nilsson, and B. Raphael, “A formal basis for the heuristic determination of minimum cost paths,” *IEEE transactions on Systems Science and Cybernetics*, vol. 4, no. 2, pp. 100–107, 1968.
- [37] A. Riabov, S. Sohrabi, and O. Udrea, “New algorithms for the top-k planning problem,” in *Proceedings of the Scheduling and Planning Applications woRKshop (SPARK) at the 24th International Conference on Automated Planning and Scheduling (ICAPS)*, 2014, pp. 10–16.
- [38] E. Karpas and C. Domshlak, “Cost-optimal planning with landmarks.”
- [39] M. Ramirez and H. Geffner, “Probabilistic plan recognition using off-the-shelf classical planners,” in *Proceedings of the Conference of the Association for the Advancement of Artificial Intelligence (AAAI 2010)*, 2010, pp. 1121–1126.

- [40] D. Bryce and S. Kambhampati, “A tutorial on planning graph based reachability heuristics,” *AI Magazine*, vol. 28, no. 1, p. 47, 2007.
- [41] V. Perera, S. P. Selvaraj, S. Rosenthal, and M. Veloso, “Dynamic Generation and Refinement of Robot Verbalization,” in *RO-MAN*, Columbia University, NY, August 2016.
- [42] A. Dragan and S. Srinivasa, “Generating legible motion,” in *Proceedings of Robotics: Science and Systems*, 2013.
- [43] A. Dragan, S. Bauman, J. Forlizzi, and S. Srinivasa, “Effects of robot motion on human-robot collaboration,” in *HRI*, 2015.
- [44] Y. Zhang, S. Sreedharan, A. Kulkarni, T. Chakraborti, H. H. Zhuo, and S. Kambhampati, “Plan Explicability and Predictability for Robot Task Planning,” in *ICRA*, 2017.
- [45] A. Kulkarni, T. Chakraborti, Y. Zha, S. G. Vadlamudi, Y. Zhang, and S. Kambhampati, “Explicable Robot Planning as Minimizing Distance from Expected Behavior,” *CoRR*, vol. abs/1611.05497, 2016.
- [46] Z. Gong and Y. Zhang, “Robot signaling its intentions in H-R teaming,” in *HRI Workshop on Explainable Robotic Systems*, 2018.
- [47] T. Chakraborti, S. Sreedharan, and S. Kambhampati, “Balancing Explanations and Explicability in Human-Aware Planning,” in *AAMAS Extended Abstract*, 2018, also appeared in AAAI 2017 Fall Symposium on Artificial Intelligence for Human-Robot Interaction (AI-HRI).
- [48] S. Sreedharan, T. Chakraborti, and S. Kambhampati, “Handling model uncertainty and multiplicity in explanations via model reconciliation,” in *ICAPS*, 2017.
- [49] T. Chakraborti, S. Sreedharan, S. Grover, and S. Kambhampati, “Plan Explanations as Model Reconciliation – An Empirical Study,” *ArXiv e-prints*, Feb. 2018.